7

LOGIC GATES AND CIRCUITS

Fundamental Gates

A logic gate is an electronic circuit which makes logic decisions. Logic 1 and 0 are represented by voltage levels.

Total No. of possible combinations of binary inputs to a gate

$N = 2^n$

N = No. of possible input combinations

n = No. of input variables (A, B, C etc.)

The Inverter (NOT gate)

A gate that performs the mathematical operations of taking the complement.

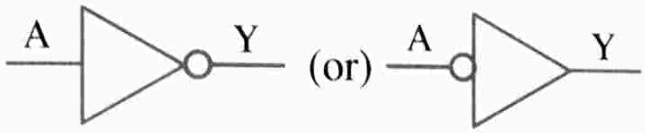
Logic symbol 

Fig. 7.1

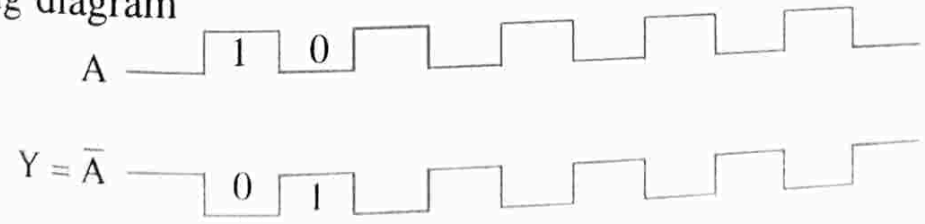
It has single input (A) and single output (Y)

Truth Table

Input A	Output Y
0	1
1	0

Boolean equation for NOT is $\boxed{Y = \bar{A}}$

Timing diagram



Law :

$\overline{0} = 1$

$\overline{1} = 0$

if $A = 0$, $\overline{A} = 1$

if $A = 1$, $\overline{A} = 0$

$\overline{\overline{A}} = A$

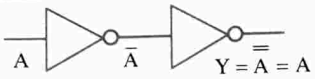


Fig. 7.2

AND gate

Two or more inputs

Single output

Fig. 7.3

All the inputs must be high for getting high output.

Truth table (for 2 inputs A and B)

No. if combinations (rows) = $2^N = 2^2 = 4$

Inputs		Ouput
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

Boolean equation $Y = A \cdot B$

Laws	
Null law	$A \cdot 0 = 0$
Identity law	$A \cdot 1 = A$
	$A \cdot A = A$
	$A \cdot \overline{A} = 0$
Commutative law	$A \cdot B = B \cdot A$

Note: Boolean multiplication is the same as the AND function. For 3 inputs (A, B, C), $2^3 = 8$ combinations. For 4 inputs (A, B, C, D), $2^4 = 16$ combinations

Timing diagram

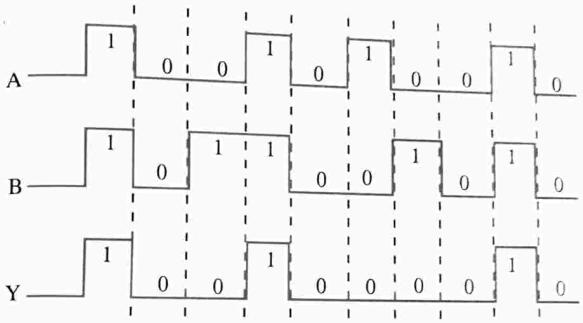


Fig. 7.4

Example : Write the output expression for the logic circuit.

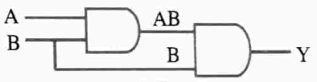


Fig. 7.5

Ans : Y = ABB

1. *Explain AND function with a 2-input AND gate* *(CU Nov. 19)*

OR gate

Two or more inputs.

Single output

$Y = A + B$

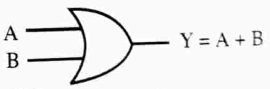


Fig. 7.6

A HIGH on the output is produced when any of the inputs are HIGH. The output is LOW only when all of the inputs are LOW.

Truth Table

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

2. *Output of an OR gate is zero only when the inputs are ——.* *(CU Nov. 2017)*

Ans : Zeros

Laws

$$A + 0 = A$$
$$A + 1 = 1$$
$$A + A = A$$
$$A + \bar{A} = 1$$

$$A + B = B + A$$

3. *Write the output expression for the large circuit & draw a timing diagram*

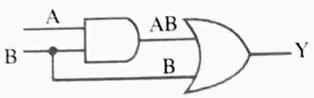


Fig. 7.7

Ans : $Y = AB + B$

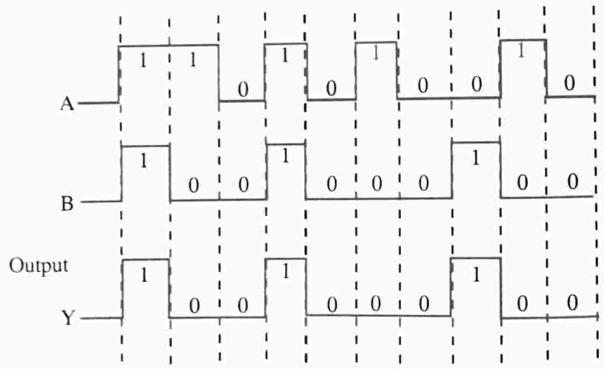


Fig. 7.8

NAND gate

NAND stands for NOT-AND. A NOT followed by an AND gate. NAND operation is the complement of the AND operation.

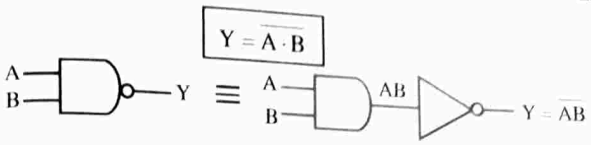


Fig. 7.9

Truth Table

Inputs		$Y = \overline{AB}$
A	B	
0	0	1
0	1	1
1	0	1
1	1	0

4. *NAND and NOR gates are known as — gates.* *(CU Nov. 17)*

Ans : Universal. (CU Nov. 19)

Exercise

1. Show the symbol and truth table of the NAND gate.
2. Write a truth table for three input NAND gate
3. If the two waveform shown in Fig. applied to the NAND gate, sketch out the resulting output waveform.

Ans :

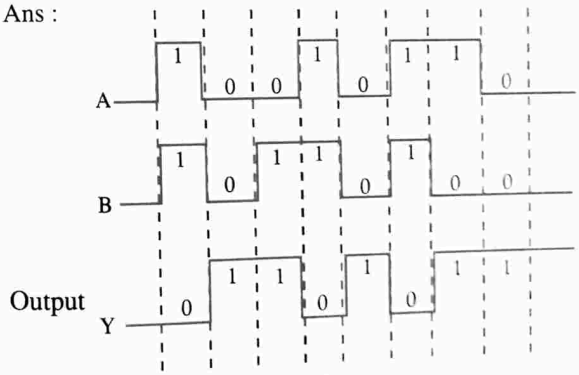
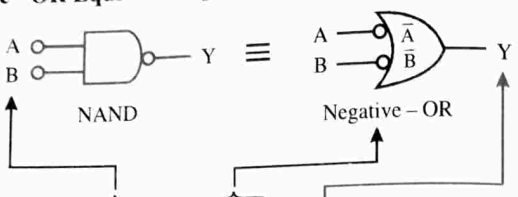


Fig. 7.10

Negative - OR Equivalent operation of a NAND gate



A	B	$\bar{A}$	$\bar{B}$	Y
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

Fig. 7.11

DE MORGAN'S THEOREMS (CU Nov. 17)

$$\overline{A+B+C} = \bar{A}\cdot\bar{B}\cdot\bar{C}$$

$$\overline{A\cdot B\cdot C} = \bar{A}+\bar{B}+\bar{C}$$

For two input variables - De Morgan's Theorem can be explained as

A	B	$\bar{A}$	$\bar{B}$	A + B	A · B	$\overline{A+B}$	$\overline{AB}$	$\bar{A}+\bar{B}$	$\bar{A}\cdot\bar{B}$
0	0	1	1	0	0	1	1	1	1
0	1	1	0	1	0	0	1	1	0
1	0	0	1	1	0	0	1	1	0
1	1	0	0	1	1	0	0	0	0

5. *State and Explain De Morgan's theorem.* *(CU Nov. 20)*

6. *Prove* $\overline{(A+B)\bar{C}\bar{D}+E+\bar{F}} = [\bar{A}\bar{B}+CD]\,\bar{E}F$ *by using De Morgan's theorem*

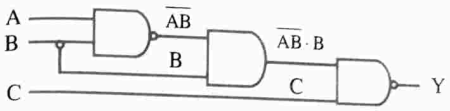
Ans : $\overline{(A+B)\bar{C}\bar{D}+E+\bar{F}} = \overline{[(A+B)\bar{C}\bar{D}]+[E+\bar{F}]}$

$$= \overline{(A+B)\bar{C}\bar{D}}\cdot\overline{(E+\bar{F})}$$

$$= \overline{[(A+B)]+[\bar{C}\bar{D}]}\cdot\bar{E}\cdot\bar{\bar{F}}$$

$$= [\bar{A}\bar{B}+CD][\bar{E}F]$$

7. *Write the output expression for the logic circuit*



$$Y = \overline{(\overline{AB}\cdot B)C}$$

$$= \overline{(\bar{A}+\bar{B})B\cdot C}$$

$$= \bar{A}BC + \bar{B}BC$$

$$= \bar{A}BC$$

Fig. 7.12

NOR gate

NOR stands for NOT-OR. A NOT followed by an OR gate. NOR operation is the complement of the OR operation.

$$\boxed{Y = \overline{A+B}}$$

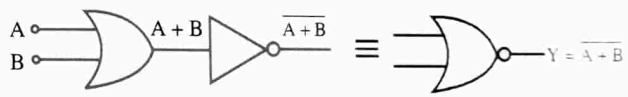


Fig. 7.13

Truth Table

Inputs		Output
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

8. *If the two waveform shown in figure are applied to a NOR gate, what is the resulting output waveform?*

Ans :

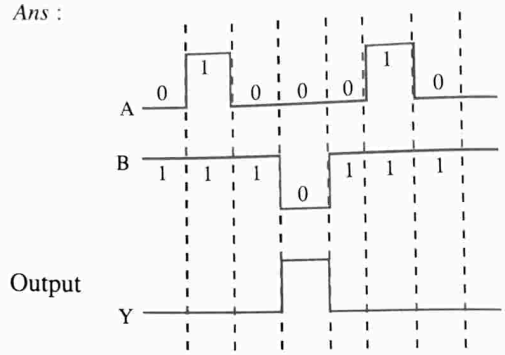


Fig. 7.14

Negative - AND equivalent operation of the NOR gate

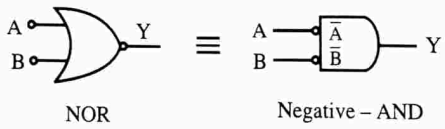


Fig. 7.15

A	B	$\bar{A}$	$\bar{B}$	Y
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0

EXCLUSIVE-OR gate (XOR)

$$Y = A \oplus B = \bar{A}B + A\bar{B}$$

Fig. 7.16

A	B	$\bar{A}B + A\bar{B}$
0	0	0
0	1	1
1	0	1
1	1	0

For an exclusive - OR gate, output Y is high when input A is LOW and input B is HIGH or when input A is HIGH and input B is LOW. Y is LOW when A and B both are HIGH or both LOW.

Equation for XOR can also be written as

$$Y = (A + B)\,(\overline{AB})$$

XOR laws:

$$A \oplus 0 = A$$
$$A \oplus 1 = \bar{A}$$
$$A \oplus B = B \oplus A$$
$$A \oplus A = 0$$
$$A \oplus \bar{A} = 1$$
$$(A \oplus B) \oplus C = A \oplus (B \oplus C)$$
$$AB \oplus AC = A(B \oplus C)$$

EXCLUSIVE - NOR Gate (X NOR)

(CU Nov. 2017)

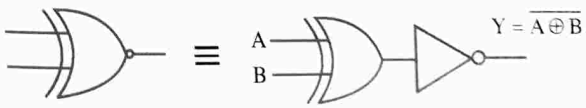


Fig. 7.17

Truth table

Inputs		Output
A	B	$Y = \overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

Duality Theorems

To convert an expression to its dual change 0 to 1 and 1 to 0. Convert AND ing to OR ing and OR ing to AND ing.

Table 7.1

Expression	Dual
$\overline{0} = 1$	$\overline{1} = 0$
$0 \cdot 1 = 0$	$1 + 0 = 1$
$0 \cdot 0 = 0$	$1 + 1 = 1$
$1 \cdot 1 = 1$	$0 + 0 = 0$
$A \cdot 1 = A$	$A + 0 = A$
$A \cdot A = A$	$A + A = A$
$A \cdot \overline{A} = 0$	$A + \overline{A} = 1$
$A \cdot B = B \cdot A$	$A + B = B + A$
$A \cdot (B \cdot C) = (A \cdot B) \cdot C$	$A + (B + C) = (A + B) + C$
$A \cdot (B + C) = AB + AC$	$A + BC = (A + B)(A + C)$
$A(A + B) = A$	$A + AB = A$
$A \cdot (A \cdot B) = A \cdot B$	$A + A + B = A + B$
$\overline{AB} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A}\,\overline{B}$
$A + B = AB + \overline{A}B + A\overline{B}$	$AB = (A + B)(\overline{A} + B)(A + \overline{B})$

9. Prove that $A + BC = (A + B)(A + C)$

$A + BC = A \cdot 1 + BC$ $(\because A \cdot 1 = A)$

$= A(1 + B) + BC$ $(\because B + 1 = 1)$

$= A \cdot 1 + AB + BC$ $(\because A(B + C) = AB + AC)$

$= A \cdot (1 + C) + AB + BC$ $(\because 1 + A = 1)$

$= A \cdot 1 + AC + AB + BC$

$= A \cdot A + AC + AB + BC$ $(\because A \cdot A = A)$

$= A(A + C) + B(A + C)$

$= (A \pm C)(A + B)$

} laws

10. Show $A + \overline{A}B = A + B$

$A + \overline{A}B = A \cdot 1 + \overline{A}B$ $(\because A \cdot 1 = A)$

$= A(1 + B) + \overline{A}B$

$= A \cdot 1 + AB + \overline{A}B$

$= A + B(A + \overline{A})$ $(\because A + \overline{A} = 1)$

$= A + B \cdot 1$ $(\because B \cdot 1 = B)$

$= A + B$

11. Show $A + AB = A$ (CU Nov. 19)

$A + AB = A(1 + B) = A \cdot 1 = A$

12. Show $A = A(A + B)$

$A(A + B) = A \cdot A + A \cdot B$

$= A + AB$ $(\because 1 + B = 1)$

$= A(1 + B)$

$= A \cdot 1 = A$

13. Prove that $AB + C = AB + BC + \overline{B}C$

$AB + BC + \overline{B}C = AB + C(B + \overline{B})$

$$= AB + C \cdot 1$$
$$= AB + C$$

14. Prove that $\bar{A} \cdot B + A \cdot B + \bar{A} \cdot \bar{B} = B + \bar{A}$

$$\text{L . H . S} = (\bar{A} + A)B + \bar{A}\bar{B}$$
$$= B + \bar{A}\bar{B} \qquad (\text{law } A + \bar{A}B = A + B)$$
$$= B + \bar{A}$$

Other laws

$A + B + C = A + (B + C)$

$A(B + C) = AB + AC$

$A \,.\, (B + C) = AB + AC$

$A + BC = (A + B)\,(A + C)$

$A + AB = A$

$A\,(A + B) = A$

$A + \bar{A}B = (A + B)$

$A\,(\bar{A} + B) = AB$

$AB + A\bar{B} = A$

$(A + B)\,(A + \bar{B}) = A$

$AB + \bar{A}C = (A + C)\,(\bar{A} + B)$

$(A + B)\,(\bar{A} + C) = AC + \bar{A}B$

$AB + \bar{A}C + BC = AB + \bar{A}C$

$(A + B)\,(\bar{A} + C)\,(B + C) = (A + B)\,(\bar{A} + C)$

15. Construct a logic circuit for the Boolean expression (A + AB) (B + BC) (C + AB) and show how it reduces to an AND gate.

Ans :

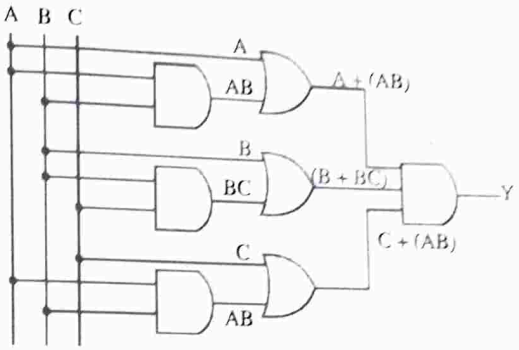


Fig. 7.18

$$Y = (A + AB)\,(B + BC)\,(C + AB)$$
$$= A\,(1 + B)\,B\,(1 + C)\,(C + AB) \qquad \because 1 + B = 1$$
$$= AB\,(C + AB) \qquad \because 1 + C = 1$$
$$= ABC + AB$$
$$= AB\,(1 + C) = AB \qquad \because 1 + C = 1$$

Fig. 7.19

16. An AOI (AND-OR-Invert) logic chip has two 4 input AND gates connected to a 2 input NOR gate. Write the Boolean expression for the circuit (assume the inputs are labled A through H)

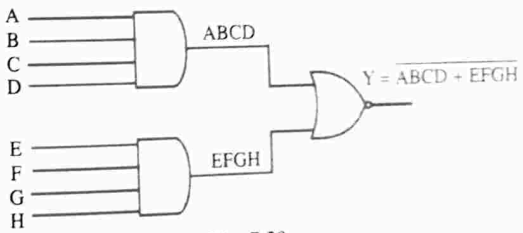


Fig. 7.20

17. *Use AND gates, OR gates and inverters as needed to implement the expression* $Y = B(C\bar{D}E + \bar{E}FG)\,(\overline{A\bar{B}} + C)$

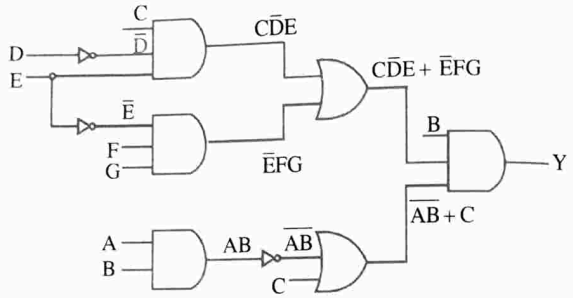


Fig. 7.21

18. *Draw the logic ckt given by the Boolean equation*

$$Y = \bar{A}BC + A\bar{B}C + AB\bar{C} + \bar{A}\bar{B}\bar{C}$$

Ans :

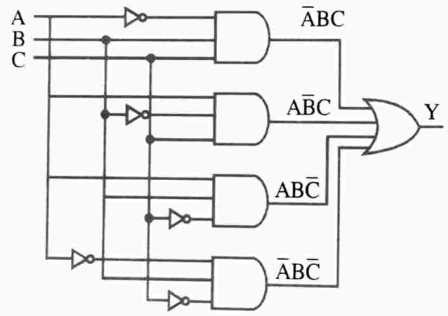


Fig. 7.22

19. *Implement logic expression* $Y = (A + B)\,(\bar{A} + \bar{B})$ *in a logic diagram. Construct the truth table and hence show that the logic diagram is equivalent to an XOR gate.*

(***Note*** : Draw back from the output Y)

Ans :

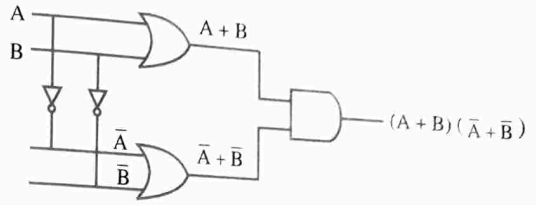


Fig. 7.23

A	B	$\bar{A}$	$\bar{B}$	$(A + B)(\bar{A} + \bar{B})$	XOR $Y = \bar{A}B + \bar{B}A$
0	0	1	1	0	0
0	1	1	0	1	1
1	0	0	1	1	1
1	1	0	0	0	0

20. *What is the importance of De Morgan's theorems in Boolean Algebra and what are the advantages of Boolean theorems?* *(CU Nov. 19)*

Exercise

4. Show the symbol and truth table of the XOR gate.

Problem

21. $Y = (A + BC)\,(B + \bar{C}A)$

(a) Design a logic circuit using the above Boolean equation.

(b) Is it possible to design the circuit with only NAND or only NOR gates? Design it.

(c) Simplify the equation. Design according to this

Ans : (a)

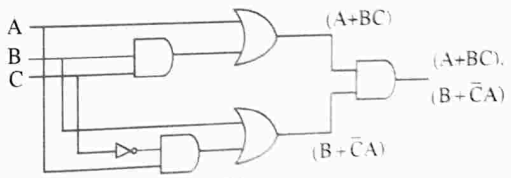


Fig. 7.24

(b) If we use the equation in SOP (sum of product) form we can contruct logic circuit using NAND gates only.

i.e, $Y = (A + BC)(B + \bar{C}A)$

$= A(B + \bar{C}A) + (BC)(B + \bar{C}A)$

$= AB + A\bar{C}A + BCB + BC\bar{C}A$ ... (1)

here $A\bar{C}A = (A \cdot A) \cdot \bar{C} = A\bar{C}$

$BCB = (B \cdot B) \cdot C = BC$

$BC\bar{C}A = B \cdot (\bar{C} \cdot C) \cdot A = B \cdot 0 \cdot A$

$= 0$

Substituting (1)

$Y = AB + A\bar{C} + BC$ (SOP)

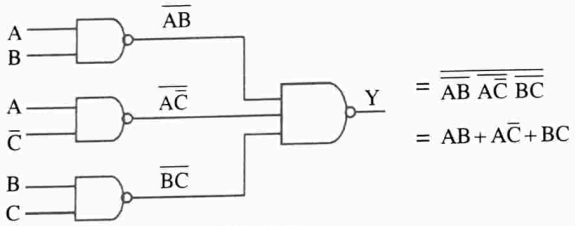


Fig. 7.25

(c) $Y = AB + A\bar{C} + BC = BC + A\bar{C}$

(using theorem $AB + \bar{A}C + BC = AB + \bar{A}C$)

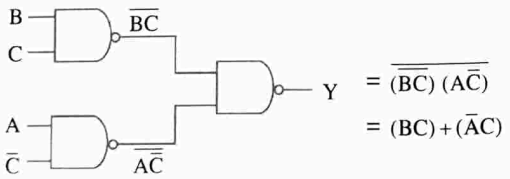


Fig. 7.26

22. *Simplify the expression* $Y = ABC + BC$ *(CU Nov. 19)*

Hints : $BC(A+1) = BC.1 = BC$

Basic Combinational Logic Circuits

AND - OR Logic

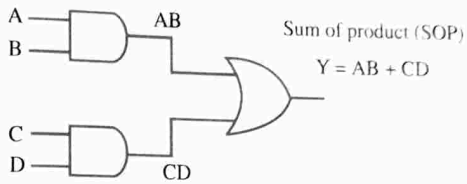
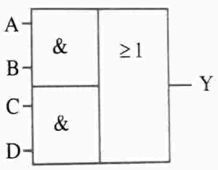


Fig. 7.27

The circuit consists of 2 input AND gates and one two input OR gate.

Its American National Standards Institute (ANSI) symbol is as follows.



Problem

23. *Three different tanks are placed at same height. Water is stored in them. A level sensor in each tank produces a HIGH voltage when the level of water in the tank drops below a particular point. Design a circuit that monitors the water level in each tank and indicates when the level in any two of the tanks drops below the specific point.*

Ans : Here AND-OR ckt is to be used.

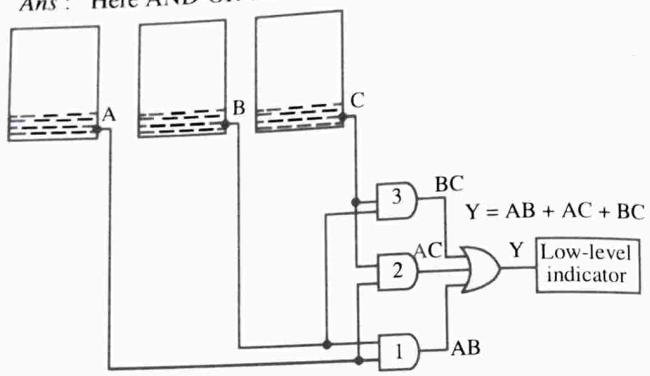


Fig. 7.28

AND gate 1 checks the levels in tanks A and B.
AND gate 2 checks the levels in tanks A and C
AND gate 3 checks the levels in tanks B and C

When the water level in any two of the tanks gets too low, one of the AND gates will have HIGHs on both of its inputs. So its output becomes HIGH. For eg:- if the levels in A and C are too low, the inputs of AND gate 2 becomes high and its output becomes HIGH. So the output Y of the OR gate also becomes HIGH. This high output glows a LED indicator.

The Boolean Some Of Product (SOP) expression for the above logic circuit is

$$Y = AB + AC + BC$$

Note: AND - OR logic directly implements SOP expressions.

AND - OR - Invert logic (AOI)

After complementing or inverting (using NOT) of an AND - OR circuit, the result is an AND - OR invert circuit.

For example : $Y = (\bar{A} + \bar{B})(\bar{C} + \bar{D})$ is a product of Sum (POS) expression.

We have, $\bar{A} + \bar{B} = \overline{AB}$ & $\bar{C} + \bar{D} = \overline{CD}$

$\therefore\ Y = (\overline{AB})(\overline{CD})$

$= \overline{\overline{(AB)}\,\overline{(CD)}}$

But $\overline{PQ} = \bar{P} + \bar{Q}$ (law)

$\therefore\ \overline{(AB)}\,\overline{(CD)} = \overline{AB} + \overline{CD}$

$\therefore\ Y = \overline{\overline{AB} + \overline{CD}}$ ($\because$ two NOTs)

$= \overline{AB + CD}$

The logic circuit for the above is

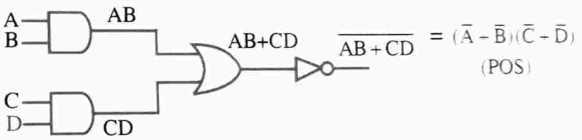


Fig. 7.29

Note: AND-OR invert logic implements (or its output is) a POS

Problem

24. *Three different tanks are placed at same height. Water is stored in them. A level sensor in each tank produces a LOW voltage when the level of water in the tank drops below a particular point. Design a ckt that monitor the water level in each tank and indicates when the level in any two of the tanks drops below the critical point.*

Ans : Here AND-OR invert circuit is used.

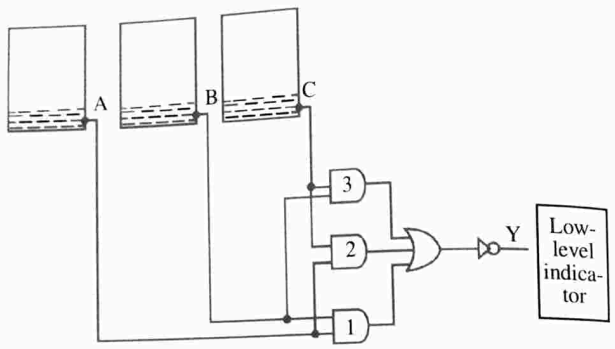


Fig. 7.30

$$Y = \overline{AB + AC + BC}$$

$$= (\bar{A} + \bar{B})(\bar{A} + \bar{C})(\bar{B} + \bar{C})$$

This gives POS

25. *Write the Boolean expression for the AND-OR invert logic in the above figure and show that the output is HIGH when any two of the inputs A, B or C are LOW.*

Ans : $Y = \overline{BC} = \bar{B} + \bar{C}$

Exercise

A water pump is required to turn on automatically whenever the water level in any two or more of the three tanks A, B, and C falls below a pre-set level. Each water tank is provided with a level detector that generates a high voltage whenever the water level in the tank is low.

(a) Write down the Boolean expression for the working of water pump whenever a high voltage turn on.

(b) Implement the Boolean expression using AND, OR and NOT gates.

(c) Reduce the Boolean expression to minimum number of literals and

(d) Obtain a much simpler circuit. (CU Nov. 2016)

Exclusive - OR logic

Logic diagram

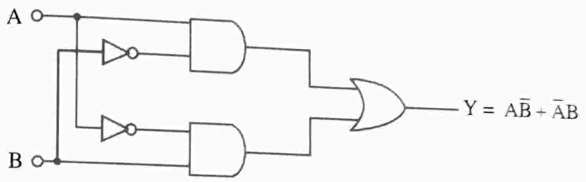


Fig. 7.31

Exclusive - NOR logic

Simply invert the output of XOR

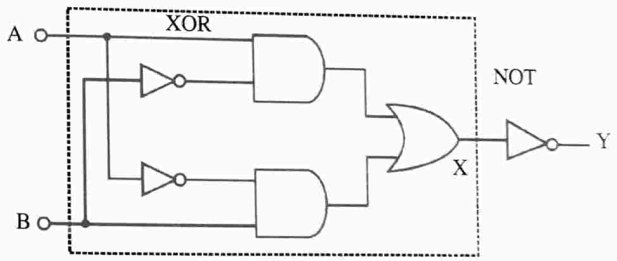


Fig. 7.32

$$X = A\bar{B} + \bar{A}B$$

$$Y = \overline{A\bar{B} + \bar{A}B}$$

$$= (\overline{A\bar{B}})\,(\overline{\bar{A}B}) \qquad (\because \text{De Morgans theorem})$$

$$= (\bar{A} + \bar{\bar{B}})\,(\bar{\bar{A}} + \bar{B})$$

$$= (\bar{A} + B)\,(A + \bar{B})$$

$$= \bar{A}\bar{B} + AB$$

26. Construct EX-OR gate by using NAND gates

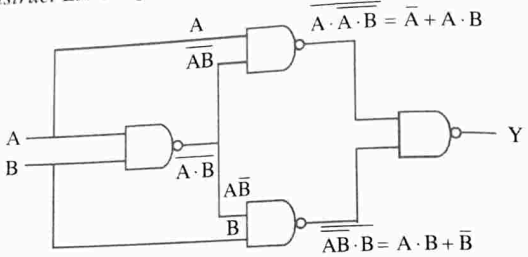


Fig. 7.33

$$Y = \overline{(\overline{A} + A \cdot B) + A \cdot B + \overline{B}}$$

$$= \overline{\overline{A}}\,\overline{AB} + \overline{AB} \cdot \overline{\overline{B}}$$

$$= A(\overline{A} + \overline{B}) + B(\overline{A} + \overline{B})$$

$$= A\overline{A} + A\overline{B} + B\overline{A} + B\overline{B}$$

$$= 0 + A\overline{B} + \overline{A}B + 0$$

$$= A\overline{B} + \overline{A}B$$

Exercise

5. Construct EX-OR gate by using NOR gates
6. Implement the logic gate for the given expression

$$Y = AB + \overline{B}CD + \overline{C}\overline{D}$$

27. Draw the block diagram and truth table of Exclusive OR gate *(CU Nov. 18)*

The Universal Property of NAND and NOR gates

It is possible to implement any logic expression using only NAND gates or NOR gates. So we can construct all other basic gates (OR, AND, NOT etc) by using either only NAND gates or only by using NOR gates. So they are called universal gates.

NAND as a universal gate

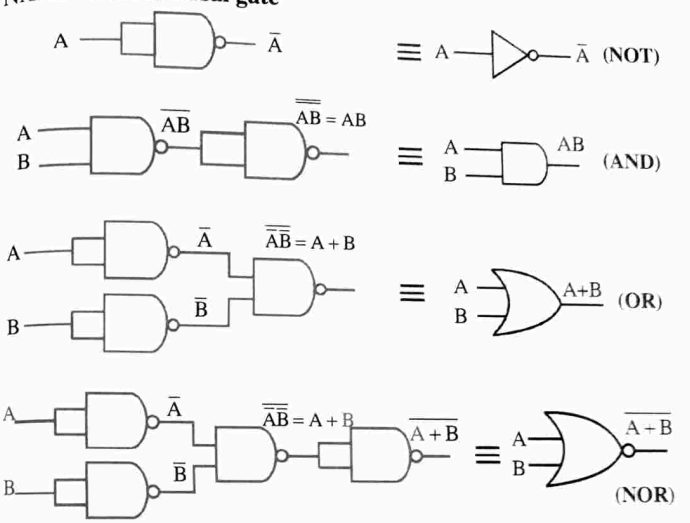


Fig. 7.34

NOR as a Universal gate

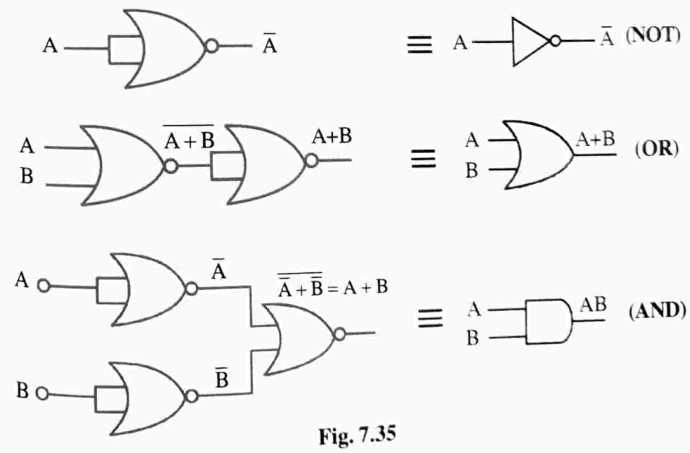


Fig. 7.35

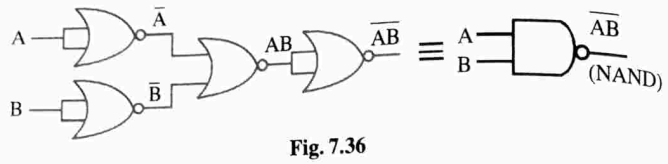
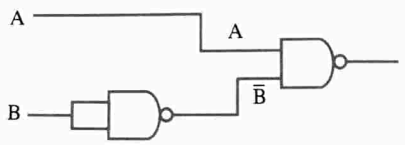


Fig. 7.36

Problems

28. *Use NAND gates to implement the expression*

$$Y = \bar{A} + B$$



$$Y = \overline{A \cdot \bar{B}}$$
$$= \bar{A} + \bar{\bar{B}}$$
$$= \bar{A} + B$$

Fig. 7.37

29. *Use NAND gates to implement* $Y = A + \bar{B}$

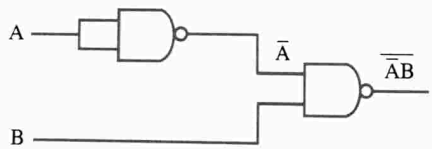
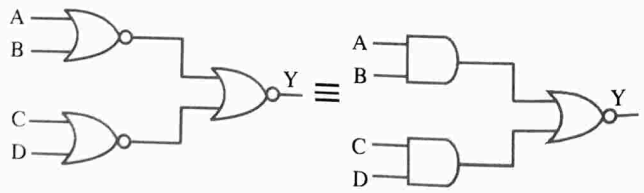


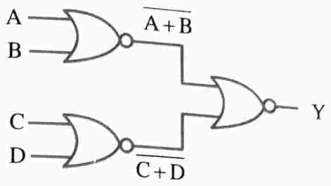
Fig. 7.38

30. *Prove that the following two circuits are logically equivalent.*



Ans : L.H.S.

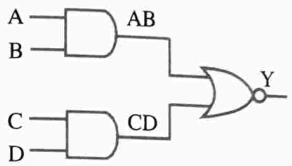
Fig. 7.39



$$= \overline{\overline{(A+B)} + \overline{(C+D)}}$$
$$= \overline{(\bar{A} \cdot \bar{B}) + (\bar{C} \cdot \bar{D})}$$
$$= \overline{(\bar{A} \cdot \bar{B})} \cdot \overline{(\bar{C} \cdot \bar{D})}$$

Fig. 7.40

R.H.S. $= AB \cdot CD$



$$Y = \overline{AB + CD}$$
$$= AB \cdot CD$$

Fig. 7.41

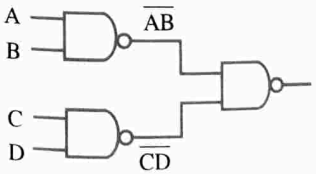
Combinational Logic using NAND and NOR gates:

NAND logic

A NAND gate can act as either a NAND gate or a Negative OR gate.

By De Morgan's theorem,

$$\underbrace{\overline{AB}}_{\text{NAND}} = \underbrace{\bar{A} + \bar{B}}_{\text{Negative-OR}}$$



$$Y = \overline{(\overline{AB})(\overline{CD})}$$
$$= \overline{(\bar{\bar{A}} + \bar{\bar{B}})(\bar{\bar{C}} + \bar{\bar{D}})}$$
$$= \overline{(\bar{\bar{A}} + \bar{\bar{B}})} + \overline{(\bar{\bar{C}} + \bar{\bar{D}})}$$
$$= \bar{\bar{A}}\bar{\bar{B}} + \bar{\bar{C}}\bar{\bar{D}}$$
$$= AB + CD$$

Fig. 7.42

Problems

31. Using only NAND gates realise the expression

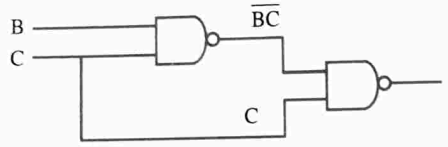
$$Y = ABC + \bar{A}BC + B\bar{C} + \bar{C}$$

Ans : Given $Y = ABC + \bar{A}BC + B\bar{C} + \bar{C}$

$= ABC + B\bar{C} + \bar{C}$

$= BC(A+1) + \bar{C}$ $\quad \because A + 1 = 1$

$= BC + \bar{C}$



$Y = \overline{\overline{BC} \cdot \bar{C}}$

$= \overline{\overline{BC} \cdot \bar{C}}$

$= BC + \bar{C}$

Fig. 7.43

32. Using only NAND gates simplify the expression

$$X = (A + \bar{B} + C)(\bar{A} + B + \bar{C})(A + \bar{B})$$

Ans : $X = (A + \bar{B} + C)(\bar{A} + B + \bar{C})(A + \bar{B})$

$= (A\bar{A} + AB + A\bar{C} + \bar{B}\bar{A} + \bar{B}B + \bar{B}\bar{C} +$

$C\bar{A} + CB + C\bar{C})\,(A + \bar{B})$

$= (0 + AB + A\bar{C} + \bar{B}\bar{A} + 0 + \bar{B}\bar{C} + C\bar{A} + CB + 0)\,(A + \bar{B})$

$= AAB + AA\bar{C} + A\bar{B}\bar{A} + A\bar{B}\bar{C} + AC\bar{A} + ACB + \bar{B}AB$

$+ \bar{B}A\bar{C} + \bar{B}\bar{B}\bar{A} + \bar{B}\bar{B}\bar{C} + \bar{B}C\bar{A} + \bar{B}CB$

$= AB + A\bar{C} + 0 + A\bar{B}\bar{C} + 0 + ABC + 0 +$

$\bar{B}A\bar{C} + \bar{B}\bar{A} + \bar{B}\bar{C} + \bar{B}C\bar{A} + 0$

$= AB + A\bar{C}(1 + \bar{B}) + \bar{B}\bar{A}(1 + C) + \bar{B}\bar{C}(A + 1) + ABC$

$= AB(1 + C) + A\bar{C} + \bar{B}\bar{A} + \bar{B}\bar{C}$

$X = AB + A\bar{C} + \bar{A}\bar{B} + \bar{B}\bar{C}$

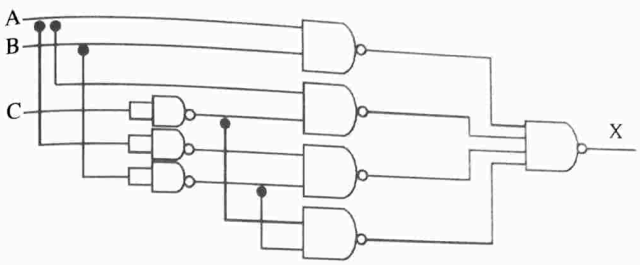


Fig. 7.44

33. Construct a logic circuit using only NAND gates for a Boolean expression.

$$Y = (\bar{A} + \bar{B})C + (\bar{D} + \bar{E})F$$

Ans :

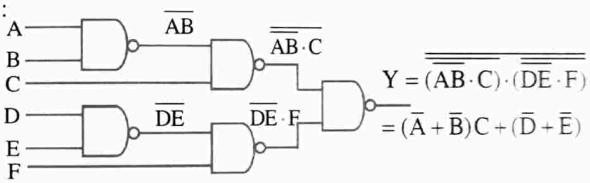


Fig. 7.45

NOR logic

A NOR gate can act as either a NOR gate or a negative - AND gate.

By Demorgan's theorem,

$$\underbrace{\overline{A + B}}_{\text{NOR gate}} = \underbrace{\bar{A}\bar{B}}_{\text{Negative AND}}$$

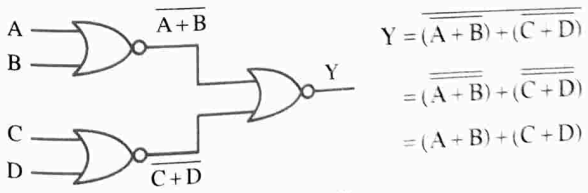


Fig. 7.46

Problem

34. $X = A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C} + \bar{B}\bar{C} + A\bar{C} = \bar{C}(\bar{B} + A)$

Arrive this result by realizing using only NOR gates.

Ans : $X = A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C} + \bar{B}\bar{C} + A\bar{C}$

$= \bar{B}\bar{C}(A + \bar{A}) + \bar{B}\bar{C} + A\bar{C}$

$= \bar{B}\bar{C} + \bar{B}\bar{C} + A\bar{C}$

$= \bar{B}\bar{C} + A\bar{C}$

$= \bar{C}(\bar{B} + A)$

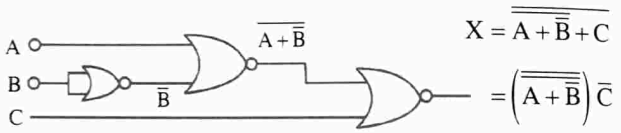


Fig. 7.47

Exercises

7. Prove $\overline{P + \overline{QR}} = 0$, using DeMorgan's theorem.

$\overline{P + \overline{QR}} = (\bar{P} \cdot QR)(P\bar{Q} + PQR)$

$= \bar{P}QR\ P\bar{Q} + \bar{P}\ QR \cdot PQR$

$= 0 + 0 = \mathbf{0}$

8. Why NAND and NOR gates are called **universal gates**?

   Ans: They can perform all the basic logic functions such as AND, OR and NOT.

9. Explain **Minterm** and **Maxterm** for three variables.

   Ans: If the input variables are A, B and C, then Minterm is $\bar{A}\bar{B}\bar{C}$ and Maxterm is A + B + C

10. Draw figures describing how NAND and NOR gates are used as invertors?

11. What do you mean by an active low input?

    Ans: If 0 is giving, it works as 1.

    If 1 is giving, it works as 0.

12. Realize XOR gate using only NAND gates.

13. Realize XOR gate using only NOR gates.

14. Realize AND gate using NAND and NOR gates.

15. Realize OR gate using NAND and NOR gates.

16. Implement the logic expression $Y = (A + B)(A' + B')$ in a logic diagram. Construct the truth table and hence show that the logic diagram is equivalent to an XOR gate. (CU Nov. 2015)

17. Why NAND and NOR gates are called universal gates?

    Ans : Using only NAND or NOR gates all other gates can be performed.

18. What are universal gates? Explain the working of NAND and NOR gates as universal gates (CU Nov. 2015)

Half Adder (HA)

Half Adder is a logic circuit for the addition of two one-bit binary numbers. Half adder gives a sum (s) and carry (c) on its output.

The symbol for Half Adder is

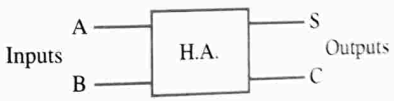


Fig. 7.48

The logic circuit for HA is

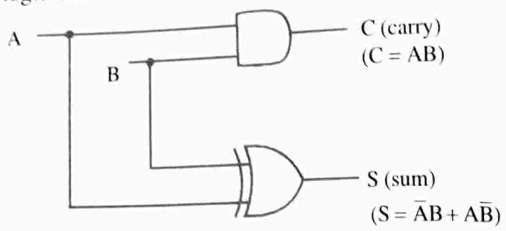


Fig. 7.49

A and B are inputs. The carry (c) output is produced with an AND gate. The Sum (s) output is produced with an Exclusive – OR gate.

The truth table for HA is as follows.

Table 7.2

Inputs		outputs	
A	B	S ($S = \bar{A}B + A\bar{B}$)	C ($C = AB$)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

35. *Discuss the operation of a Half Adder* *(CU Nov. 15)*
36. *Using NAND gates construct an HA.*

Ans :

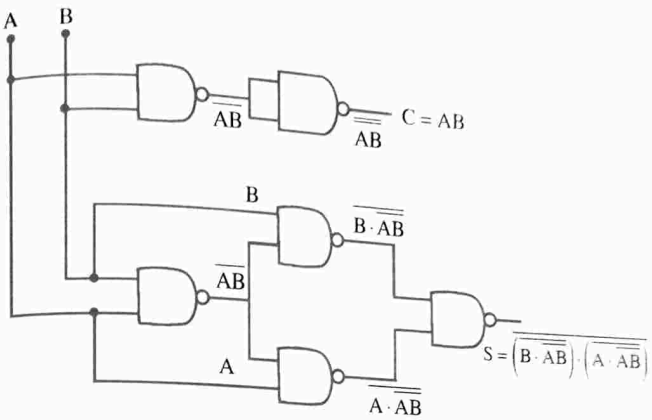


Fig. 7.50

$$S = \overline{\left(\overline{B \cdot \overline{AB}}\right) \cdot \left(\overline{B \cdot \overline{AB}}\right)} \quad \text{using De Morgan's theorem}$$

$$= \overline{\overline{B \cdot \overline{AB}}} + \left(\overline{\overline{A \cdot \overline{AB}}}\right)$$

$$= B \cdot \overline{AB} + A \cdot \overline{AB}$$

$$= B(\bar{A} + \bar{B}) + A(\bar{A} + \bar{B})$$

$$= B\bar{A} + B\bar{B} + A\bar{A} + A\bar{B}$$

$$= A\bar{B} + B\bar{A}$$

37. *Draw the block diagram of a half adder and write down its truth table* *(CU Nov. 20)*

Full Adder (FA)

Full adder is a logic circuit that can add three bits at a time. There are three inputs. Sum and carry are outputs. The additional input used here is for handling input carries.

When we want to add two binary numbers, each having two or more bits, the LSBs can be added by using a HA. The carry resulted from the addition of LSBs is carried over to the next column and added to the two bits in that colum.

The block diagram or symbol is

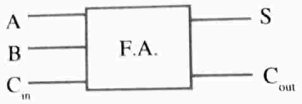


Fig. 7.51

The logic circuit of a FA is as follows

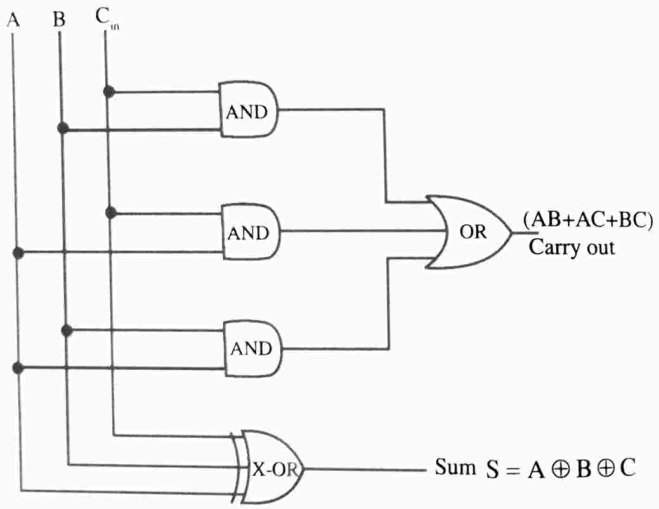


Fig. 7.52

Truth table of FA is as follows

Table 7.3

Inputs			Outputs	
A	B	Cin	Sum S	Carry C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Here A, B, C_{in} are the three inputs. (C_{in} represents any carry generated by the previous stage). Sum and carry (out) are the outputs. C_{out} is the carry output to be added in to the next stage.

Sum $S = A \oplus B \oplus C$

$= \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$

Carry output $C_{out} = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$

$= \bar{A}BC + A\bar{B}C + AB(\bar{C} + C)$

$= \bar{A}BC + A\bar{B}C + AB \quad [\because \bar{C} + C = 1]$

$= \bar{A}BC + A(\bar{B}C + B)$

$= \bar{A}BC + A(B + C)$

$= \bar{A}BC + AB + AC = B(\bar{A}C + A) + AC$

$= B(A + C) + AC \quad [\because \bar{A}C + A = A + C]$

$C_{out} = \mathbf{AB + BC + AC}$

38. Construct a FA by using two HA and an OR gate

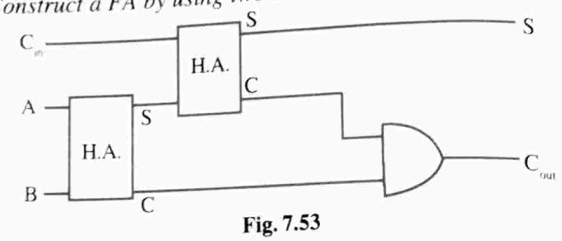


Fig. 7.53

39. Draw the block diagram of a full adder and write down its truth table. *(CU Nov. 18)*

Flip Flops (FF)

A FF is a bistable electronic circuit which has two stable states. It can remain in either of the states indefinitely. Its state can be changed by applying the proper triggering signal. It is made up of an assembly of logic gates. Its symbol is

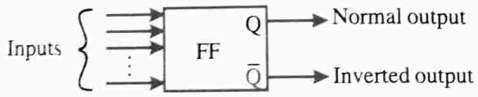


Fig. 7.54

R – S Flip Flop

Two cross coupled NOR gates form R – S Flip Flop.

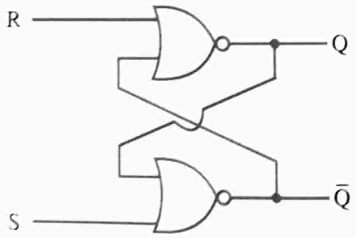


Fig. 7.55

R and S are two inputs. Q and $\overline{Q}$ are outputs. Output of each gate is connected to the input of the other gate. (This produces the feedback).

Truth table for a NOR gate R.S Flip Flop is as shown below:

Table 7.4

R	S	Q	Action
0	0	Last value	No change
0	1	1	Set (S)
1	0	0	Reset (R)
1	1	?	unpredictable (Forbidden)

40. Draw the timing diagram for an RS Flip Flop.

Ans :

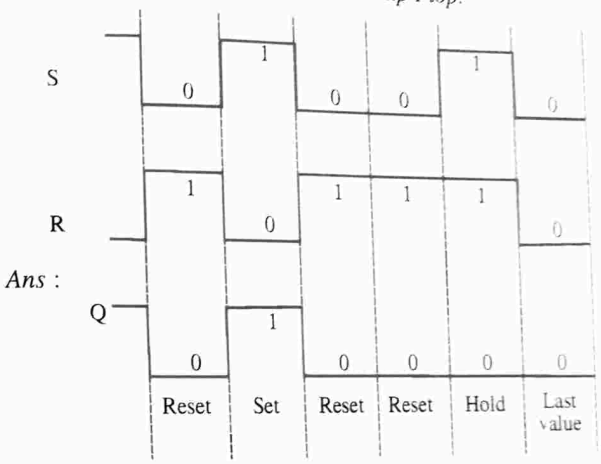


Fig. 7.56

41. With the help of diagrams explain the working of RS flip-flop. *(CU Nov. 18, CU Nov. 20)*

JK Flip Flop

JK Flip Flop is a refinement of RS Flip Flop. But the intermediate state (when $R = S = 1$) of the RS type is defined in the JK type. In that condition the state of the output is changed. (i.e., the complement of the previous state is available. For example if the previous state of the output Q is 0, it becomes 1 and vice versa).

Here the inputs J and K behave like inputs S and R to set and reset the Flip Flop. (J is for set and K is for reset). J and K are called control inputs. Because they determine what the Flip Flop does when a +ve clock edge arrives.

The logic symbol for JK Flip Flop is

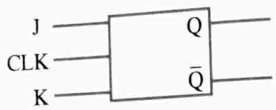


Fig. 7.57

Its logic diagram is as follows

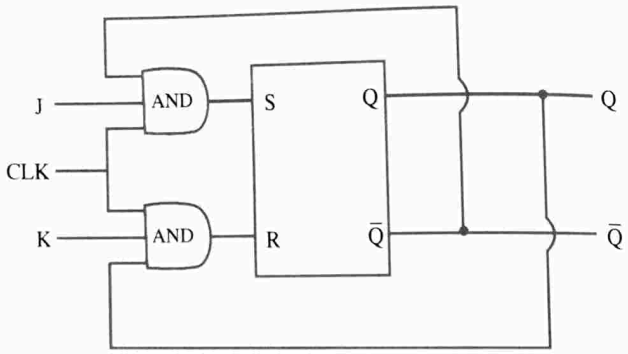


Fig. 7.58

The RS Flip Flop is converted in to JK Flip Flop by making $S = J \cdot \bar{Q}$ and $R = K \cdot Q$

Because of AND gates, the above circuit is +ve edge triggered (i.e., FF changes state at +ve edge of the clock pulse)

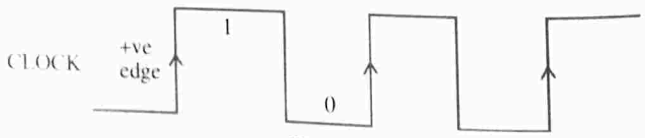


Fig. 7.59

(Note that J and K are adjacent letters in alphabets. There is no other meaning for them. But S is for set and R is for Reset).

Truth table for JK Flip Flop is as shown below

Table 7.5

Inputs			Outputs	
CLK	J	K	Q	$\bar{Q}$
0	0	0	**Hold (No change)** clock is at 0 state. ∴ FF is not working. Q retains at its last value	
0	0	1		
0	1	0		
0	1	1		
1	0	0	**Hold** ∵ J = K = 0 even though CLK is high	
1	0	1	0	1
1	1	0	1	0
1	1	1	Toggle	

When clock input is at LOW level the data inputs have no effect on outputs.

When J = 0, K = 0, the FF remains in hold state eventhough the CLK is HIGH.

When CLK is HIGH, J = 0, K = 1 the output Q is in reset (cleared to 0). Here the output Q is ANDed with K and CLK inputs. So FF is cleared.

When CLK is HIGH J = 1, K = 1, the repeated clock pulses cause the output to turn OFF, ON, ON, OFF .... (toggle).

42. *Discuss the functioning of a JK Flip Flop with the help of a logic circuit* (CU Nov. 15)

Use : in registers

43. *With the help of diagrams explain the working of RS and JK flip flops.* (CU Nov. 17)